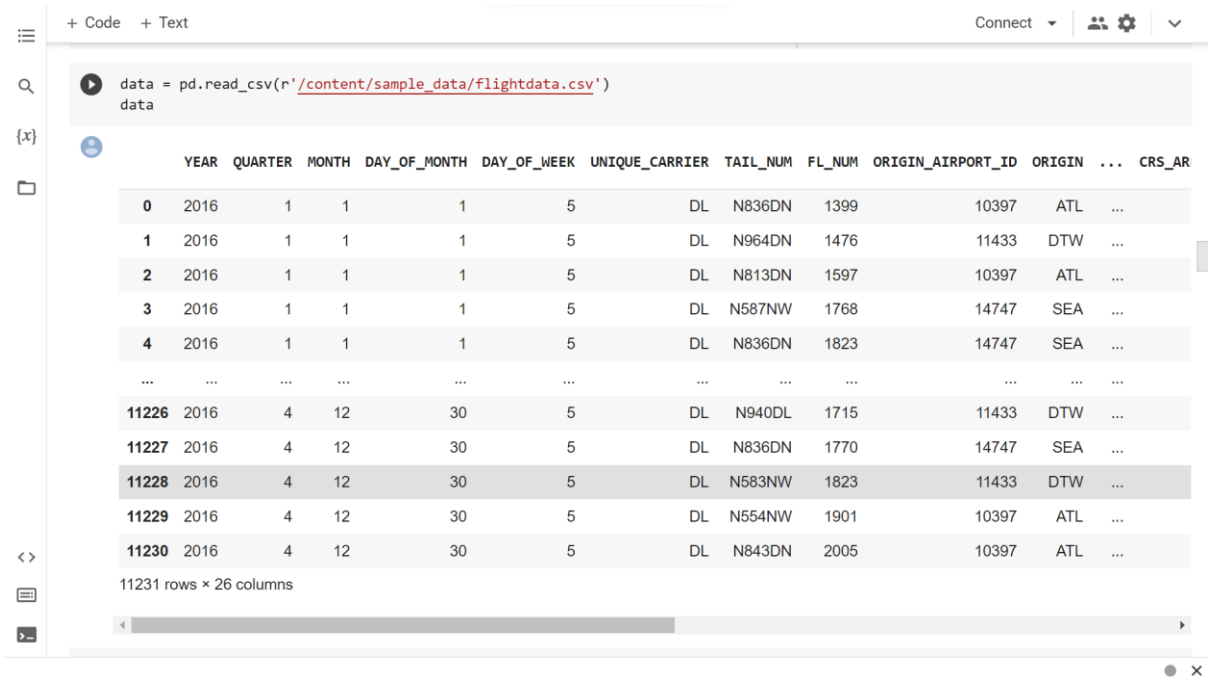
1.

```
import pandas as pd
import numpy as np
import pickle
import matplotlib.pyplot as pit
%matplotlib inline
import seaborn as sns
import sklearn
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import GradientBoostingClassifier,RandomForestClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import RandomizedSearchCV
import imblearn
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix, f1_score
```

2.



	YEAR	QUARTER	MONTH	DAY_OF_MONTH	DAY_OF_WEEK	UNIQUE_CARRIER	TAIL_NUM	FL_NUM	ORIGIN_AIRPORT_ID	ORIGIN	...	CRS_AR
0	2016	1	1	1	5	DL	N836DN	1399	10397	ATL	...	
1	2016	1	1	1	5	DL	N964DN	1476	11433	DTW	...	
2	2016	1	1	1	5	DL	N813DN	1597	10397	ATL	...	
3	2016	1	1	1	5	DL	N587NW	1768	14747	SEA	...	
4	2016	1	1	1	5	DL	N836DN	1823	14747	SEA	...	
...	...	...	...	...	...	...	...	...	...	...	...	...
11226	2016	4	12	30	5	DL	N940DL	1715	11433	DTW	...	
11227	2016	4	12	30	5	DL	N836DN	1770	14747	SEA	...	
11228	2016	4	12	30	5	DL	N583NW	1823	11433	DTW	...	
11229	2016	4	12	30	5	DL	N554NW	1901	10397	ATL	...	
11230	2016	4	12	30	5	DL	N843DN	2005	10397	ATL	...	

11231 rows × 26 columns

3.

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11231 entries, 0 to 11230
Data columns (total 26 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   YEAR                 11231 non-null  int64
 1   QUARTER              11231 non-null  int64
 2   MONTH                11231 non-null  int64
 3   DAY_OF_MONTH         11231 non-null  int64
 4   DAY_OF_WEEK          11231 non-null  int64
 5   UNIQUE_CARRIER       11231 non-null  object
 6   TAIL_NUM             11231 non-null  object
 7   FL_NUM               11231 non-null  int64
 8   ORIGIN_AIRPORT_ID    11231 non-null  int64
 9   ORIGIN               11231 non-null  object
 10  DEST_AIRPORT_ID      11231 non-null  int64
 11  DEST                 11231 non-null  object
 12  CRS_DEP_TIME         11231 non-null  int64
 13  DEP_TIME             11124 non-null  float64
 14  DEP_DELAY            11124 non-null  float64
 15  DEP_DEL15            11124 non-null  float64
 16  CRS_ARR_TIME         11231 non-null  int64
 17  ARR_TIME             11116 non-null  float64
 18  ARR_DELAY            11043 non-null  float64
 19  ARR_DEL15            11043 non-null  float64
 20  CANCELLED            11231 non-null  float64
 21  DIVERTED             11231 non-null  float64
 22  CRS_ELAPSED_TIME     11231 non-null  float64
 23  ACTUAL_ELAPSED_TIME  11043 non-null  float64
 24  DISTANCE             11231 non-null  float64
 25  Unnamed: 25          0 non-null      float64
dtypes: float64(12), int64(10), object(4)
memory usage: 2.2+ MB
```

4.

```
import math

for index, row in data. iterrows():
    data.loc[index, 'CRS_ARR_TIME'] = math.floor ( row['CRS_ARR_TIME']/100)
data.head()
```

	YEAR	QUARTER	MONTH	DAY_OF_MONTH	DAY_OF_WEEK	UNIQUE_CARRIER	TAIL_NUM	FL_NUM	ORIGIN_AIRPORT_ID	ORIGIN	...	CRS_ARR_TIME	ARR_TIME	ARR_DELAY	ARR_DEL15	CANCELLED	DI
0	2016	1	1	1	5	DL	N836DN	1399	10397	ATL	...	21	2102.0	-41.0	0.0	0.0	
1	2016	1	1	1	5	DL	N964DN	1476	11433	DTW	...	14	1439.0	4.0	0.0	0.0	
2	2016	1	1	1	5	DL	N813DN	1597	10397	ATL	...	12	1142.0	-33.0	0.0	0.0	
3	2016	1	1	1	5	DL	N587NW	1768	14747	SEA	...	13	1345.0	10.0	0.0	0.0	
4	2016	1	1	1	5	DL	N836DN	1823	14747	SEA	...	6	615.0	8.0	0.0	0.0	

5 rows × 26 columns

5.

```
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
data['DEST'] = le.fit_transform(data['DEST'])
data['ORIGIN'] = le.fit_transform(data['ORIGIN'])
data.head(5)
```

	YEAR	QUARTER	MONTH	DAY_OF_MONTH	DAY_OF_WEEK	UNIQUE_CARRIER	TAIL_NUM	FL_NUM	ORIGIN_AIRPORT_ID	ORIGIN	...	CRS_ARR_TIME	ARR_TIME	ARR_DELAY	ARR_DEL15	CANCELLED	DIVE
)	2016	1	1	1	5	DL	N836DN	1399	10397	0	...	21	2102.0	-41.0	0.0	0.0	
I	2016	1	1	1	5	DL	N964DN	1476	11433	1	...	14	1439.0	4.0	0.0	0.0	
?	2016	1	1	1	5	DL	N813DN	1597	10397	0	...	12	1142.0	-33.0	0.0	0.0	
)	2016	1	1	1	5	DL	N587NW	1768	14747	4	...	13	1345.0	10.0	0.0	0.0	
I	2016	1	1	1	5	DL	N836DN	1823	14747	4	...	6	615.0	8.0	0.0	0.0	

rows × 26 columns

6.

```
[ ] data['ORIGIN'].unique()
```

```
array([0, 1, 4, 3, 2])
```

```
x=data.iloc[:,0:8].values
y=data.iloc[:,8:9].values
x
```

```
array([[2016, 1, 1, ..., 'DL', 'N836DN', 1399],
       [2016, 1, 1, ..., 'DL', 'N964DN', 1476],
       [2016, 1, 1, ..., 'DL', 'N813DN', 1597],
       ...,
       [2016, 4, 12, ..., 'DL', 'N583NW', 1823],
       [2016, 4, 12, ..., 'DL', 'N554NW', 1901],
       [2016, 4, 12, ..., 'DL', 'N843DN', 2005]], dtype=object)
```

7.

```
[ ] from sklearn.preprocessing import OneHotEncoder
    oh = OneHotEncoder()
    z=oh. fit_transform(x[:,4:5]).toarray()
    t=oh. fit_transform(x[:,5:6]).toarray()
    #x=np.delete(x,[4,7],axis=1)
```

```
[ ] z
```

```
array([[0., 0., 0., ..., 1., 0., 0.],
       [0., 0., 0., ..., 1., 0., 0.],
       [0., 0., 0., ..., 1., 0., 0.],
       ...,
       [0., 0., 0., ..., 1., 0., 0.],
       [0., 0., 0., ..., 1., 0., 0.],
       [0., 0., 0., ..., 1., 0., 0.]])
```

```
t
```

```
array([[1.],
       [1.],
       [1.],
       ...,
       [1.],
       [1.],
       [1.]])
```

8.

```
[ ] x=np.delete(x,[4,5],axis=1)
```

```
data.describe()
```

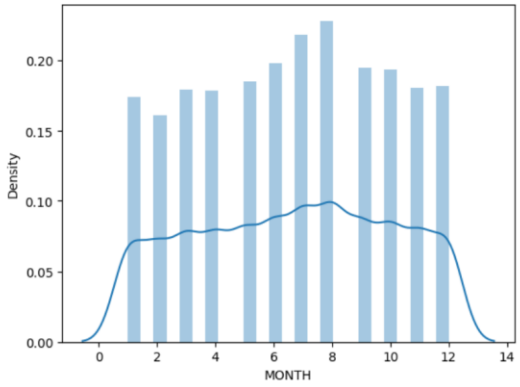
	YEAR	QUARTER	MONTH	DAY_OF_MONTH	DAY_OF_WEEK	FL_NUM	ORIGIN_AIRPORT_ID	ORIGIN	DEST_AIRPORT_ID	DEST	...	CRS_ARR_TIME
count	11231.0	11231.000000	11231.000000	11231.000000	11231.000000	11231.000000	11231.000000	11231.000000	11231.000000	11231.000000	...	11231.000000
mean	2016.0	2.544475	6.628973	15.790758	3.960199	1334.325617	12334.516695	1.837325	12302.274508	1.806607	...	15.067314
std	0.0	1.090701	3.354678	8.782056	1.995257	811.875227	1595.026510	1.489464	1601.988550	1.496328	...	5.023534
min	2016.0	1.000000	1.000000	1.000000	1.000000	7.000000	10397.000000	0.000000	10397.000000	0.000000	...	0.000000
25%	2016.0	2.000000	4.000000	8.000000	2.000000	624.000000	10397.000000	0.000000	10397.000000	0.000000	...	11.000000
50%	2016.0	3.000000	7.000000	16.000000	4.000000	1267.000000	12478.000000	2.000000	12478.000000	2.000000	...	15.000000
75%	2016.0	3.000000	9.000000	23.000000	6.000000	2032.000000	13487.000000	3.000000	13487.000000	3.000000	...	19.000000
max	2016.0	4.000000	12.000000	31.000000	7.000000	2853.000000	14747.000000	4.000000	14747.000000	4.000000	...	23.000000

8 rows × 24 columns

9.

```
sns.distplot(data.MONTH)
```

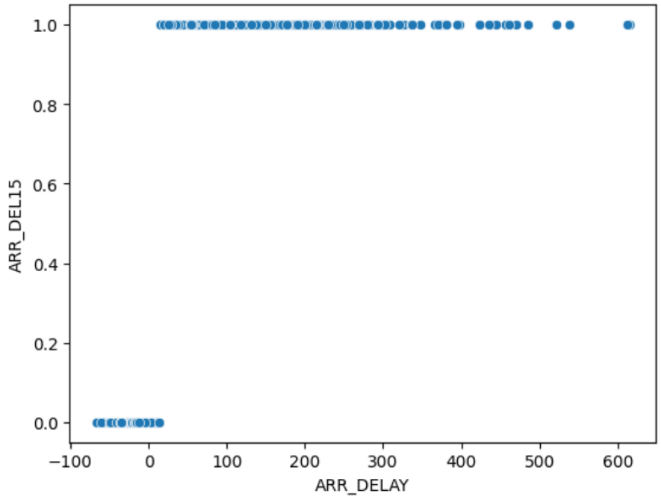
```
<ipython-input-25-45ea5d7e7464>:1: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(data.MONTH)
<Axes: xlabel='MONTH', ylabel='Density'>
```



10.

```
sns.scatterplot(x='ARR_DELAY',y='ARR_DEL15',data=data)
```

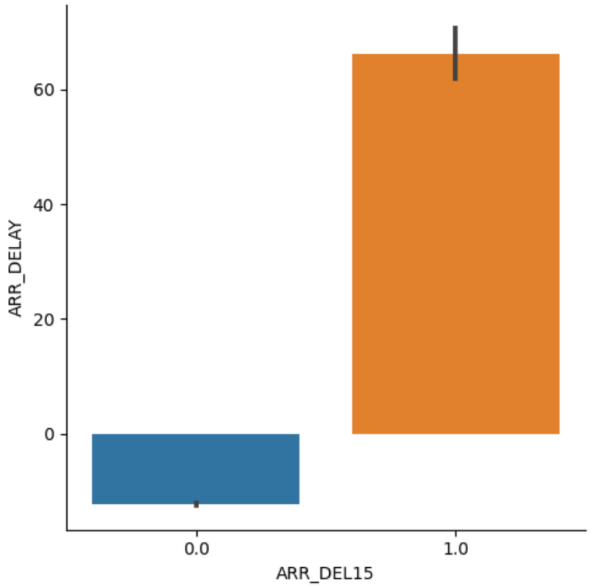
```
<Axes: xlabel='ARR_DELAY', ylabel='ARR_DEL15'>
```



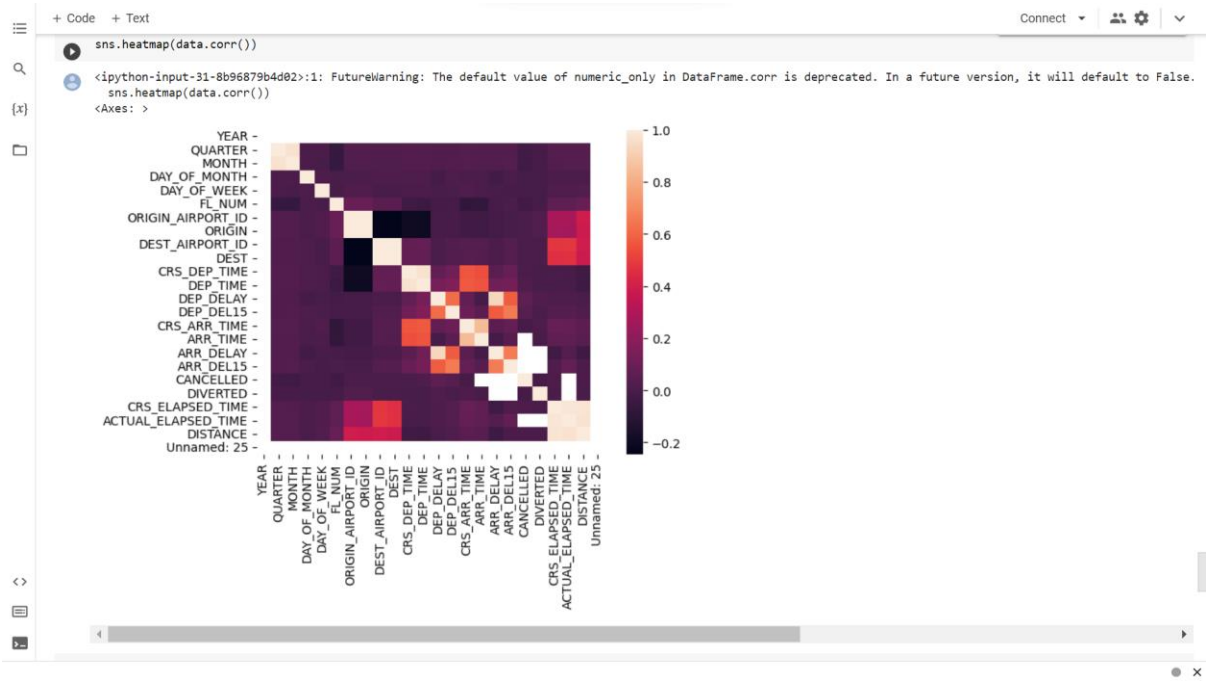
11.

```
sns.catplot(x="ARR_DEL15",y="ARR_DELAY",kind='bar',data=data)
```

```
<seaborn.axisgrid.FacetGrid at 0x7f1d637b0370>
```



12.



13.

```
[ ] data = pd.get_dummies(data,columns=['ORIGIN','DEST'])
    data.head()
```

	YEAR	QUARTER	MONTH	DAY_OF_MONTH	DAY_OF_WEEK	UNIQUE_CARRIER	TAIL_NUM	FL_NUM	ORIGIN_AIRPORT_ID	DEST_AIRPORT_ID	...	ORIGIN_0	ORIGIN_1	ORIGIN_2	ORIGIN_
0	2016	1	1	1	5	DL	N836DN	1399	10397	14747	...	1	0	0	(
1	2016	1	1	1	5	DL	N964DN	1476	11433	13487	...	0	1	0	(
2	2016	1	1	1	5	DL	N813DN	1597	10397	14747	...	1	0	0	(
3	2016	1	1	1	5	DL	N587NW	1768	14747	13487	...	0	0	0	(
4	2016	1	1	1	5	DL	N836DN	1823	14747	11433	...	0	0	0	(

5 rows × 34 columns

```
x=data.iloc[:,0:8].values
y=data.iloc[:,8:9].values
```

14.

```
[ ] from sklearn.model_selection import train_test_split
    x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.2,random_state=0)
```

```
[ ] x_test.shape
```

```
(2247, 8)
```

```
x_train.shape
```

```
(8984, 8)
```

```
[ ] y_test.shape
```

```
(2247, 1)
```

```
[ ] y_train.shape
```

```
(8984, 1)
```